

*iTCO*

---

Sun Solaris™:  
Security Features

---

**White Paper**

## Introduction

This document describes four levels of securing Solaris™ using features native to the operating system:

- Level 1 consists of features and tools that help administrators tightly control who can log onto the system.
- Level 2 describes tools that enable administrators to set the overall security state of the system.
- Level 3 covers Secure Distributed services and Developers Platforms, describing how Solaris supports different authentication and encryption mechanisms.
- Level 4 describes the tools to control access to the physical network.

## Security: A Key Issue for the Global Corporation

The fact that stand-alone data centers with fully centralized security requirements are fading rapidly from modern corporate computing environments is well documented. More cost effective and efficient distributed environments in which client systems are separated from servers over a network continue to proliferate. In addition, interconnections between commercial, private, and government institutions worldwide are expanding the community of users who can potentially gain access to internal corporate resources.

Concurrently, users are becoming more knowledgeable and sophisticated. Unfortunately, some have used their knowledge for unscrupulous purposes. Although famous "hackers" continue to grab media headlines, studies have shown that a great majority of computer crimes go undetected. These trends have all given rise to fundamental changes in the security requirements for the global corporation.

It is no wonder that security has emerged as a key issue for companies wishing to capitalize on the benefits of implementing global distributed systems, without risking the privacy and integrity of sensitive information. System and network administrators therefore must be able to choose products that offer a full spectrum of features that address their changing security needs.

## Level 1: Controlling Login Access on Solaris

The first level of Solaris security control consists of features and tools that help administrators tightly control who can log in to the system. Central to this purpose is the use of a password which can be used to check the identity of the person who is attempting to log in. In other words, if one assumes the password is known only to its owner, it can be used to prove that the individual who is attempting to log in is indeed "authentic" and is acknowledged to have been granted the "right" to log in to the system.

Unfortunately, the system is unable to determine if the password is being used by someone other than its owner. It is for this reason that protecting passwords is of the utmost importance. Solaris has many features that control the creation, use and safe storage of passwords. These are known collectively as password management features and they include the following:

- Password validation: Solaris compares the password the user provides to the one set by and stored for that user in a special file (see "Shadow Password File" below). If the passwords match, the user is allowed to log in.
- Password aging: Solaris enables the administrator to set an expiration date for passwords. Solaris will warn the user that the password is about ready to expire and ask for a new one to be set. Once the password expires, if a new one hasn't been set, a log in attempt will be refused. (Note that NIS+ on Solaris 2.5 includes support for domain wide password aging.)
- Disallow old password: The longer a password is around, the more likely it is that someone will be able to find out what it is. This feature prevents a user from reusing a previously used password.
- Password qualification: Solaris helps ensure that you will create a password that will be more difficult for someone else to guess. It does this by checking to see if the password has the correct number of characters and/or symbols.
- Shadow password file: This "hidden" file (called /etc./shadow) stores all users' passwords and is readable only by root. Before the implementation of the shadow password file, other privileged users were able to access the password file.
- Account expiration: This enables a system administrator to set an expiration date for an account. After the date expires, the account is no longer active.

The Security Manager offers products for Solaris that impose further login restrictions. Some examples include:

- Restricting hours of access: This means certain users are not allowed to log in at times that are pre-determined by the system administrator, for example in the middle of the night when no one is around.
- Disable login on repeated invalid attempts: This prevents users (and programs) from trying to guess passwords by repeatedly attempting to log in.
- Autolockscreen and logout: After a prescribed period of idle time, a workstation will automatically lockscreen or log out.
- Increased controls over root/su privilege: Requires one to supply special passwords to access root and super user privileges.

## ***Remote Access Protection***

Since potentially unwanted system accesses can also be attempted over remote dialup lines, Solaris enables modem ports to be password protected. Once a modem port password is set, a user who dials up from a remote location will be

first asked to provide it. If the correct one is supplied, the user will be allowed to enter the formal system login process.

Additional products that provide remote access protection are available, like digital token technology. A digital token is a "one time only" password that is used to control remote dial-in access. A set of pre-assigned digital tokens are stored in a hand held device that can be carried to remote locations. In this scheme passwords are never reused, resulting in tighter security.

## **Level 2: System Resource Access Control**

Once a user is successfully logged in, they can begin to attempt to access resources. Therefore, Solaris enables administrators to control the general accessibility of resources on the system by providing tools which can set the overall security state of the system. Solaris also offers features that enable file access privileges to be set appropriately. In addition, auditing capability is provided to aid in tracking access attempts. These features are described in detail below.

### ***Setting and Checking the Security State of Solaris***

Most system administrators would agree that there are times when it is essential to try to assess the overall security state of the system and/or to set it appropriately. To address this, Solaris includes the Automated Security Access Tool (ASET). ASET can automatically assess the state of the system as well as to place it in one of three pre-determined security states: low, medium, or high.

When run periodically, ASET will alert the administrator to any potential security breaches. Below is a sample of what ASET checks for:

- Existence of a system EEPROM password which protects an unauthorized individual from booting the system in single user mode
- Insecure use of the UMASK variable which dictates the default setting for file permissions when a file is created
- Insecure use of the PATH variable which outlines the order in which directories will be searched for a specified executable command or program
- System file permission settings
- Existence of new setuid programs
- Home directory permissions
- Contents of: `.rhosts`, `/etc/passwd`, `/etc/group`
- Size of files in `/usr/bin` and `/bin`

Note that administrators have the option of being alerted to potential problems by electronic mail.

When used to set the system in low security mode, ASET ensures that file attributes (permissions) are set at the standard release values. Several checks are performed and any potential security weaknesses are reported.

The medium security setting provides adequate security for most environments. ASET will modify permission settings of some system files (e.g. `ttytab`, `host.equiv`) and parameters to restrict system access. Additional security checks are performed and weaknesses and access modifications are reported.

The high security setting produces an exceedingly secure system. Many system files and parameters are set to enable only minimum access. The high security setting also can disable IP forwarding. However, this functionality has been superceded by a product called Solstice™ FireWall-1™ or Solstice™ SunScreen™, covered later in this document.

With ASET, an administrator need not waste valuable time and effort manually "hunting" for security holes on the system. ASET is also a unique feature on Solaris and is not available on other system platforms.

## ***File Protection on Solaris***

Files represent a major resource that must be protected on any system platform from PCs to mainframes. Solaris implements two methods for file protection: traditional "Unix-style" permission settings and Access Control Lists (ACLs). With Unix-style permission setting, it is possible to set read, write, and execute permission indications for a file's owner, selected groups, or the "world" (also known as "other"). However, one disadvantage of permission setting is that access can only be restricted on a per-group basis and cannot single out special privilege (or restriction) for individuals.

Support for Access Control Lists (ACLs) is in Solaris. Access Control Lists are just what the name implies: a list that controls access to files. With ACLs, extensive lists of authorization information can be maintained for every file, enabling a finer granularity of control over file access. For example, with ACLS access can be controlled on a per-user basis in addition to on a per group basis.

ACLs on Solaris are compliant with the POSIX 1003.6 specification. They have been implemented for both the User File System (UFS) as well as for NFS Versions 2 and Version 3.

## ***Auditing***

Auditing is included in this section because it helps administrators track security related events including many different types of access attempts. If a violation occurs, an audit log can help determine what happened and perhaps even help discover who the culprit is! Solaris includes two methods for auditing: Unix System Logs and C2 auditing. Both are discussed in this section

Unix system logs (syslogs) keep track of login events, resource usage, quotas, and more. Many system facilities use syslogs to record or alert the system administrator to important events. Shell scripts, or wrappers, can also be written to syslog databases to cover specific situations.

C2 Auditing, also called Controlled Access Protection, can produce a more detailed audit report. The Department of Defense defined C2 auditing as part of its guidelines for various levels of computer security in the 1980's. These requirements are outlined in the Orange Book or Trusted Computer Systems Evaluation Criteria (TCSEC). Security levels are listed starting with D for the lowest, up to A1 for the highest. The National Computer Security Center (NCSC) evaluates systems based on this criteria.

C2 can create audit logs by user, event and class. In addition, with C2 it is possible to log any event that a system administrator deems security "relevant". Solaris C2 auditing includes the Basic Security Mode (BSM) functionality, which enables the logging of events down to the system call level.

In the unlikely event that there is a security infraction, Solaris auditing capability ensures system administrators a detailed account of relevant activity. This information can be vital in helping to track down the source of the problem.

## **Level 3: Secure Distributed Services and Developer Platforms**

The Solaris core operating environment incorporates the ONC+™ family of distributed services which can optionally be configured to run with additional security features enabled. When this is the case, ONC+ consists of the Secure NIS+ distributed naming service, the Secure NFST distributed file service, and the Secure Transport Independent Remote Procedure Call (TI-RPC) platform (also known simply as Secure RPC) for building distributed applications and services.

The DCE family of services in an unbundled product called DCE for Solaris. This product includes the DFS distributed file service, the CDS distributed naming service, and the DCE RPC based developer platform (as well as other features not pertinent to this document).

Both secure ONC+ and DCE services rely on foundation technology described in this section.

### ***Foundation Technology for Secure Services***

Before a user on a client system is given access to a server's resources, the server must be sure that the user is acknowledged to have "rightful" access to the server and its resources. Therefore, in this scenario the server must be able to:

1. Check a user's identity over the network. This function is provided by an authentication service and also usually includes services listed in #3 below.

2. Make sure the user is authorized to access the resources (s)he is attempting to access once (s)he has been properly authenticated. This is provided by an authorization service.
3. Maintain the privacy and integrity of the information being exchanged over the network. These are referred to as privacy and integrity services respectively.

### ***Authentication, Privacy and Integrity Services***

Authenticating a user over the network requires that sensitive "credentials" information must be exchanged between a client and server system. Since there is no way to guarantee that this information will not be intercepted as it travels to its destination, it must be protected from being interpreted or changed along the way. Thus, a secure distributed service must have a way to protect the privacy and integrity respectively of the information.

Privacy and integrity services are typically bundled with the authentication service. A privacy service provides a way to convert information into a form that can only be interpreted by the intended recipient. This is also referred to as encrypting the data. The recipient is responsible for decrypting the data or in other words converting it back into a readable form. An integrity service provides a way to compute an information checksum which when examined will indicate whether the information has been altered from its original content.

There are many authentication services both existing and emerging. Therefore Solaris supports a flexible architecture that enables access to multiple authentication mechanisms both today and in the future. Currently on Solaris, authentication mechanisms including Kerberos, Diffie-Hellman and Unix-style [1]) are supported and are accessible via the Secure TI-RPC interface. In fact, both Secure NIS+ and Secure NFS have been developed on the Secure TI-RPC platform.

Secure DCE services on Solaris access Kerberos authentication via the DCE RPC interface. Both DFS and CDS have been programmed to the DCE RPC interface and can thus utilize the Kerberos authentication service.

### ***Pluggable Authentication Module (PAM)***

PAM provides a pluggable model for system authentication mechanisms as well as for other related services such as password, account, and session management. These services are particularly useful to applications providing or requiring "system entry" (or login) that must verify user identity as well as account information. Some common examples of these applications include login, dtlogin, rlogin, rsh, telnet, ftp, etc.

The security mechanisms accessible through PAM are implemented as dynamically loadable, shared software modules that can be installed by administrators

transparently to applications. PAM enables the administrator to configure the user authentication mechanism on a per application basis. For example, a site may require S/Key password authentication for telnet access while allowing console login sessions with just UNIX password authentication. With PAM it is also possible to configure multiple authentication mechanisms for each application. For example, an administrator may want users to get authenticated by both Kerberos and RSA. Finally, PAM enables users of these applications to supply a single password even though multiple authentication services may be in use.

## ***GSS-API***

GSS-API is a proposed standard, as defined in RFC-1508 and RFC-1509. It is becoming the de-facto standard for dealing with security services (such as authentication, integrity, and encryption), in a generic, extensible fashion. Applications can be run independent of the underlying security mechanism and technologies. It also allows for source level portability.

The PAM API and GSS-API are complimentary to each other, whereas the PAM API support for user authentication by the system entry servers, GSSAPI supports network-based client/server authentication. Therefore, once users on client systems are authenticated through PAM, they can communicate securely with server systems using GSSAPI based authentication services.

## ***Authorization Services***

An authorization service or mechanism provides a way to ensure that a user has been granted permission to access the information (s)he is attempting to access remotely. NFS supports 2 authorization mechanisms: file permission indications and POSIX 1003.6 compliant ACLs. Both permissions and Access Control Lists are covered in the section entitled "File Protection on Solaris". Secure NIS+ utilizes a methodology called table access rights to indicate authorization to access information stored in NIS+ tables.

DCE DFS uses it's own ACL standard which is referred to as DCE ACLs. The DCE CDS distributed naming service makes use of the DCE ACLs as well.

## ***Secure Remote Utilities***

In addition to secure services, secure remote utilities such as telnet, ftp, rcp, rsh, and rlogin are available for Solaris. These are sometimes referred to as "Kerberized" utilities because they often utilize Kerberos authentication.

## ***Single Signon***

A key issue in multi-vendor, distributed environments has to do with the fact that, in most cases, each host or server requires the user to provide a separate password in order to gain access to services. A method called single signon is

emerging to solve this problem. With single signon, the user enters only one password to gain access to all systems in a distributed environment.

Although it is only part of the solution, the PAM interface discussed in the section entitled "Authentication, Privacy and Integrity Services" helps enable single signon capability due to it's ability to integrate multiple authentication mechanisms.

## **Level 4: Controlling Access to the Physical Network**

Early computer networks were not designed for tight security control because it was assumed that sites (and users) connected to the network were largely trustable. Time and experience have shown that this is no longer a good assumption. In addition to potential threats from outsiders, well-intentioned internal users might accidentally expose corporate data or services from within a network to the outside world. It is therefore desirable to prevent both types of problems from occurring without requiring everyone to become security experts.

### ***Solstice FireWall-1***

The purpose of a "firewall", also known as a network security system, is to ensure that all communication between the local organization's network and an external network conforms to the organization's defined network security policies. Some examples of network security policies might be "allow access to all services unless expressly denied" or "deny access to all services unless expressly allowed". Once the security policies are established, FireWall-1 can assist in implementing a network environment based on the established policies.

Solstice FireWall-1 is a combined hardware and software solution designed to allow or disallow packets from entering the internal network based on the established security policies. The external network is often a public network such as the Internet. However, FireWall-1 can also be used to control traffic between different departments within a local network as seen in Figure 4.

Solstice FireWall-1 combines features such as protocol "aware " individual packet screening with application-level and circuit gateways to provide an efficient, generic and secure packet filtering engine. In addition to filtering technology, it includes a powerful logging and alerting system to help keep abreast of attempted violations. Also included is an intuitive, object-oriented user interface which facilitates set up and configuration.

### ***Solstice SunScreen***

Solstice SunScreen combines firewall functionality with network level (or IP) authentication, also known as SKIP (Simple Key Management for IP). It is network, protocol, and application independent. The unique "stealth" architecture of SunScreen empowers organizations with the ability to set up a virtual secure private network across public network connections such as the Internet. Because it is not a router, packets pass through without recording any indications of it's existence. Solstice SunScreen is therefore undetectable, giving potential intruders less knowledge to exploit.

The SunScreen configuration consists of a central hardware device (called SunScreen SPF-100) and a secure Administration Station from which the SunScreen security rules and parameters are specified. It allows several wires to be administered as a single network with the same range of IP addresses. This reduces the need for additional IP addresses and interfaces while providing a central place for logging and administration.

*Simplify IT*  
***iTCO***  
*your eBusiness Partner*  
[www.itcosolutions.com](http://www.itcosolutions.com)

---

iTCO Solutions Corporation  
P.O. Box 610090  
Redwood City, CA 94061  
United States

<http://www.itcosolutions.com/>

Enterprise Sales Team Contact  
Ryan Edwards  
National Accounts Manger  
Tel: 650-367-0514  
E-Mail: [redwards@itcosolutions.com](mailto:redwards@itcosolutions.com)